

CLAIMS

What is claimed is:

1. A method for executing an obfuscated application program, the method comprising:

receiving an application program comprising application program data and application
program instructions;

determining an application program instruction location permutation to apply to a current
instruction counter value;

receiving said current instruction counter value;

applying said application program instruction location permutation to said current instruction
counter value to obtain a reference to an application program instruction to execute; and

executing said application program instruction to execute.
2. The method of claim 1 wherein said application program instruction location permutation
further comprises advancing said current instruction counter if there is another application
program instruction to be executed.
3. The method of claim 1, further comprising:

determining whether there is another application program instruction to be executed;

advancing said current instruction counter if there is another application program instruction
to be executed; and

repeating said receiving, said applying, and said executing after said advancing.

4. The method of claim 1 wherein said applying further comprises:

using one or more low order bits of said current instruction counter value to obtain a modifier value from an instruction location permutation table, said instruction location permutation table comprising one or more entries comprising a first value and a modifier value, said modifier value determining how said first value is modified, said first value matching said one or more low order bits; and
modifying the low order bits of a copy of said current instruction counter value based at least in part on said modifier value to obtain a reference to an instruction to execute.

5. The method of claim 4 wherein said modifying further comprises:

replacing said low order bits of said copy said modifier value; and
right-shifting said copy by a number of bits comprising said low order bits of said copy.

6. The method of claim 4 wherein said modifying further comprises adding said modifier value to said low order bits of said copy.

7. The method of claim 4 wherein said modifying further comprises subtracting said modifier value from said low order bits of said copy.

8. The method of claim 1 wherein said applying further comprises:
- determining a permutation set size (S) and an instruction location permutation table size (N),
- said permutation set comprising a bit slice of a current instruction counter value;
- using N low order bits of said permutation set to obtain a modifier value from an instruction location permutation table, said instruction location permutation table indexed by said N low order bits;
- modifying the N high order bits of the permutation set of a copy of said current instruction counter value based at least in part on said modifier value; and
- replacing the (S-N) low order bits of said permutation set of said copy with the (S-N) high order bits of said permutation set of said current instruction counter value to obtain a reference to an instruction to execute.
9. The method of claim 8 wherein said modifying further comprises replacing said N high order bits of said permutation set of said copy with said modifier value.
10. The method of claim 8 wherein said modifying further comprises adding said modifier value to said N high order bits of said permutation set of said copy.
11. The method of claim 8 wherein said modifying further comprises subtracting said modifier value from said N high order bits of said permutation set of said copy.

12. The method of claim 1 wherein said applying further comprises:

determining a permutation set size (S) and an instruction location permutation table size (N),
said permutation set comprising a bit slice of a current instruction counter value;
using N high order bits of said permutation set to obtain a modifier value from an instruction
location permutation table, said instruction location permutation table indexed by said N
high order bits;
modifying the N low order bits of the permutation set of a copy of said current instruction
counter value based at least in part on said modifier value; and
replacing the (S-N) high order bits of said permutation set of said copy with the (S-N) low
order bits of said permutation set of said current instruction counter value to obtain a
reference to an instruction to execute.

13. The method of claim 12 wherein said modifying further comprises replacing said N low
order bits of said permutation set of said copy with said modifier value.

14. The method of claim 12 wherein said modifying further comprises adding said modifier
value to said N low order bits of said permutation set of said copy.

15. The method of claim 12 wherein said modifying further comprises subtracting said modifier
value from said N low order bits of said permutation set of said copy.

16. A method for application program obfuscation, the method comprising:
- reading a first application program comprising application program data and application program instructions;
 - determining an application program instruction location permutation that transforms said first application program into an obfuscated application program, said obfuscated application program having at least one application program instruction stored at a memory location that is based at least in part on a permutation of the memory location where the corresponding application program instruction is stored in said first application program;
 - applying said application program instruction location permutation to said first application program to create an obfuscated application program; and
 - sending said obfuscated application program.
17. The method of claim 16, further comprising receiving an application program request from a user device, said determining occurring in response to said receiving.
18. The method of claim 16, further comprising generating at least one instruction to fill one or more holes in the instruction stream of said application program if said instruction stream of said application program has one or more holes.
19. The method of claim 16 wherein
- said method further comprises, after said applying said application program instruction location permutation, applying a cryptographic process to said obfuscated application

program together with a cryptographic key to create an encrypted obfuscated application program; and

said sending comprises sending said encrypted obfuscated application program.

20. A program storage device readable by a machine, embodying a program of instructions executable by the machine to perform a method for executing an obfuscated application program, the method comprising:

receiving an application program comprising application program data and application program instructions;

determining an application program instruction location permutation to apply to a current instruction counter value;

receiving said current instruction counter value;

applying said application program instruction location permutation to said current instruction counter value to obtain a reference to an application program instruction to execute; and
executing said application program instruction to execute.

21. The program storage device of claim 20 wherein said application program instruction location permutation further comprises advancing said current instruction counter if there is another application program instruction to be executed.

22. The program storage device of claim 20, said method further comprising:

determining whether there is another application program instruction to be executed;

advancing said current instruction counter if there is another application program instruction to be executed; and
repeating said receiving, said applying, and said executing after said advancing.

23. The program storage device of claim 20 wherein said applying further comprises:

using one or more low order bits of said current instruction counter value to obtain a modifier value from an instruction location permutation table, said instruction location permutation table comprising one or more entries comprising a first value and a modifier value, said modifier value determining how said first value is modified, said first value matching said one or more low order bits; and
modifying the low order bits of a copy of said current instruction counter value based at least in part on said modifier value to obtain a reference to an instruction to execute.

24. The program storage device of claim 23 wherein said modifying further comprises:

replacing said low order bits of said copy said modifier value; and
right-shifting said copy by a number of bits comprising said low order bits of said copy.

25. The program storage device of claim 23 wherein said modifying further comprises adding said modifier value to said low order bits of said copy.

26. The program storage device of claim 23 wherein said modifying further comprises subtracting said modifier value from said low order bits of said copy.
27. The program storage device of claim 20 wherein said applying further comprises:
- determining a permutation set size (S) and an instruction location permutation table size (N),
 - said permutation set comprising a bit slice of a current instruction counter value;
 - using N low order bits of said permutation set to obtain a modifier value from an instruction location permutation table, said instruction location permutation table indexed by said N low order bits;
 - modifying the N high order bits of the permutation set of a copy of said current instruction counter value based at least in part on said modifier value; and
 - replacing the (S-N) low order bits of said permutation set of said copy with the (S-N) high order bits of said permutation set of said current instruction counter value to obtain a reference to an instruction to execute.
28. The program storage device of claim 27 wherein said modifying further comprises replacing said N high order bits of said permutation set of said copy with said modifier value.
29. The program storage device of claim 27 wherein said modifying further comprises adding said modifier value to said N high order bits of said permutation set of said copy.

30. The program storage device of claim 27 wherein said modifying further comprises subtracting said modifier value from said N high order bits of said permutation set of said copy.
31. The program storage device of claim 20 wherein said applying further comprises:
- determining a permutation set size (S) and an instruction location permutation table size (N),
 - said permutation set comprising a bit slice of a current instruction counter value;
 - using N high order bits of said permutation set to obtain a modifier value from an instruction location permutation table, said instruction location permutation table indexed by said N high order bits;
 - modifying the N low order bits of the permutation set of a copy of said current instruction counter value based at least in part on said modifier value; and
 - replacing the (S-N) high order bits of said permutation set of said copy with the (S-N) low order bits of said permutation set of said current instruction counter value to obtain a reference to an instruction to execute.
32. The program storage device of claim 31 wherein said modifying further comprises replacing said N low order bits of said permutation set of said copy with said modifier value.
33. The program storage device of claim 31 wherein said modifying further comprises adding said modifier value to said N low order bits of said permutation set of said copy.

34. The program storage device of claim 31 wherein said modifying further comprises subtracting said modifier value from said N low order bits of said permutation set of said copy.
35. A program storage device readable by a machine, embodying a program of instructions executable by the machine to perform a method for application program obfuscation, the method comprising:
- reading a first application program comprising application program data and application program instructions;
- determining an application program instruction location permutation that transforms said first application program into an obfuscated application program, said obfuscated application program having at least one application program instruction stored at a memory location that is based at least in part on a permutation of the memory location where the corresponding application program instruction is stored in said first application program;
- applying said application program instruction location permutation to said first application program to create an obfuscated application program; and
- sending said obfuscated application program.
36. The program storage device of claim 35, said method further comprising receiving an application program request from a user device, said determining occurring in response to said receiving.

37. The program storage device of claim 35, said method further comprising generating at least one instruction to fill one or more holes in the instruction stream of said application program if said instruction stream of said application program has one or more holes.

38. The program storage device of claim 35 wherein

said method further comprises, after said applying said application program instruction location permutation, applying a cryptographic process to said obfuscated application program together with a cryptographic key to create an encrypted obfuscated application program; and

said sending comprises sending said encrypted obfuscated application program.

39. An apparatus for executing an obfuscated application program, the apparatus comprising:

means for receiving an application program comprising application program data and application program instructions;

means for determining an application program instruction location permutation to apply to a current instruction counter value;

means for receiving said current instruction counter value;

means for applying said application program instruction location permutation to said current instruction counter value to obtain a reference to an application program instruction to execute; and

means for executing said application program instruction to execute.

40. The apparatus of claim 39 wherein said application program instruction location permutation further comprises means for advancing said current instruction counter if there is another application program instruction to be executed.

41. The apparatus of claim 39, further comprising:

means for determining whether there is another application program instruction to be executed;

means for advancing said current instruction counter if there is another application program instruction to be executed; and

means for repeating said receiving, said applying, and said executing after said advancing.

42. The apparatus of claim 39 wherein said means for applying further comprises:

means for using one or more low order bits of said current instruction counter value to obtain a modifier value from an instruction location permutation table, said instruction location permutation table comprising one or more entries comprising a first value and a modifier value, said modifier value determining how said first value is modified, said first value matching said one or more low order bits; and

means for modifying the low order bits of a copy of said current instruction counter value based at least in part on said modifier value to obtain a reference to an instruction to execute.

43. The apparatus of claim 42 wherein said modifying further comprises:
- means for replacing said low order bits of said copy said modifier value; and
 - means for right-shifting said copy by a number of bits comprising said low order bits of said copy.
44. The apparatus of claim 43 wherein said means for modifying further comprises means for adding said modifier value to said low order bits of said copy.
45. The apparatus of claim 43 wherein said means for modifying further comprises means for subtracting said modifier value from said low order bits of said copy.
46. The apparatus of claim 39 wherein said means for applying further comprises:
- means for determining a permutation set size (S) and an instruction location permutation table size (N), said permutation set comprising a bit slice of a current instruction counter value;
 - means for using N low order bits of said permutation set to obtain a modifier value from an instruction location permutation table, said instruction location permutation table indexed by said N low order bits;
 - means for modifying the N high order bits of the permutation set of a copy of said current instruction counter value based at least in part on said modifier value; and

means for replacing the (S-N) low order bits of said permutation set of said copy with the (S-N) high order bits of said permutation set of said current instruction counter value to obtain a reference to an instruction to execute.

47. The apparatus of claim 46 wherein said means for modifying further comprises means for replacing said N high order bits of said permutation set of said copy with said modifier value.

48. The apparatus of claim 47 wherein said means for modifying further comprises adding said modifier value to said N high order bits of said permutation set of said copy.

49. The apparatus of claim 47 wherein said means for modifying further comprises means for subtracting said modifier value from said N high order bits of said permutation set of said copy.

50. The apparatus of claim 39 wherein said means for applying further comprises:

means for determining a permutation set size (S) and an instruction location permutation table size (N), said permutation set comprising a bit slice of a current instruction counter value;

means for using N high order bits of said permutation set to obtain a modifier value from an instruction location permutation table, said instruction location permutation table indexed by said N high order bits;

means for modifying the N low order bits of the permutation set of a copy of said current instruction counter value based at least in part on said modifier value; and

means for replacing the (S-N) high order bits of said permutation set of said copy with the (S-N) low order bits of said permutation set of said current instruction counter value to obtain a reference to an instruction to execute.

51. The apparatus of claim 50 wherein said means for modifying further comprises means for replacing said N low order bits of said permutation set of said copy with said modifier value.

52. The apparatus of claim 50 wherein said means for modifying further comprises means for adding said modifier value to said N low order bits of said permutation set of said copy.

53. The apparatus of claim 50 wherein said means for modifying further comprises means for subtracting said modifier value from said N low order bits of said permutation set of said copy.

54. An apparatus for application program obfuscation, the apparatus comprising:

means for reading a first application program comprising application program data and application program instructions;

means for determining an application program instruction location permutation that transforms said first application program into an obfuscated application program, said

obfuscated application program having at least one application program instruction stored at a memory location that is based at least in part on a permutation of the memory location where the corresponding application program instruction is stored in said first application program;

means for applying said application program instruction location permutation to said first application program to create an obfuscated application program; and
means for sending said obfuscated application program.

55. The apparatus of claim 54, further comprising means for receiving an application program request from a user device, said determining occurring in response to said receiving.

56. The apparatus of claim 54, further comprising means for generating at least one instruction to fill one or more holes in the instruction stream of said application program if said instruction stream of said application program has one or more holes.

57. The apparatus of claim 54 wherein said apparatus further comprises:

means for applying a cryptographic process to said obfuscated application program together with a cryptographic key to create an encrypted obfuscated application program after said applying said application program instruction location permutation; and
means for sending said encrypted obfuscated application program.

58. An apparatus for executing an obfuscated application program, the apparatus comprising a user device configured to:
- receive an application program comprising application program data and application program instructions;
 - determine an application program instruction location permutation to apply to a current instruction counter value;
 - receive said current instruction counter value;
 - apply said application program instruction location permutation to said current instruction counter value to obtain a reference to an application program instruction to execute; and
 - execute said application program instruction to execute.
59. The apparatus of claim 58 wherein said application program instruction location permutation is configured to advance said current instruction counter if there is another application program instruction to be executed.
60. The apparatus of claim 58 wherein said user device is further configured to:
- determine whether there is another application program instruction to be executed;
 - advance said current instruction counter if there is another application program instruction to be executed; and
 - repeat said receiving, said applying, and said executing after said advancing.

61. The apparatus of claim 58 wherein said user device is further configured to apply said application program instruction location permutation by:
- using one or more low order bits of said current instruction counter value to obtain a modifier value from an instruction location permutation table, said instruction location permutation table comprising one or more entries comprising a first value and a modifier value, said modifier value determining how said first value is modified, said first value matching said one or more low order bits; and
- modifying the low order bits of a copy of said current instruction counter value based at least in part on said modifier value to obtain a reference to an instruction to execute.
62. The apparatus of claim 61 wherein said user device is further configured to modify said current instruction counter value by::
- replacing said low order bits of said copy said modifier value; and
- right-shifting said copy by a number of bits comprising said low order bits of said copy.
63. The apparatus of claim 61 wherein said user device is further configured to modify said current instruction counter value by adding said modifier value to said low order bits of said copy.

64. The apparatus of claim 61 wherein said device is further configured to modify said current instruction counter value by subtracting said modifier value from said low order bits of said copy.
65. The apparatus of claim 58 wherein said user device is further configured to apply said instruction location by:
- determining a permutation set size (S) and an instruction location permutation table size (N),
 - said permutation set comprising a bit slice of a current instruction counter value;
 - using N low order bits of said permutation set to obtain a modifier value from an instruction location permutation table, said instruction location permutation table indexed by said N low order bits;
 - modifying the N high order bits of the permutation set of a copy of said current instruction counter value based at least in part on said modifier value; and
 - replacing the (S-N) low order bits of said permutation set of said copy with the (S-N) high order bits of said permutation set of said current instruction counter value to obtain a reference to an instruction to execute.
66. The apparatus of claim 65 wherein said device is further configured to modify said current instruction counter value by replacing said N high order bits of said permutation set of said copy with said modifier value.

67. The apparatus of claim 65 wherein said device is further configured to modify said current instruction counter value by adding said modifier value to said N high order bits of said permutation set of said copy.
68. The apparatus of claim 65 wherein said device is further configured to modify said current instruction counter value by subtracting said modifier value from said N high order bits of said permutation set of said copy.
69. The apparatus of claim 58 wherein said user device is further configured to apply said instruction location permutation by:
- determining a permutation set size (S) and an instruction location permutation table size (N),
 - said permutation set comprising a bit slice of a current instruction counter value;
 - using N high order bits of said permutation set to obtain a modifier value from an instruction location permutation table, said instruction location permutation table indexed by said N high order bits;
 - modifying the N low order bits of the permutation set of a copy of said current instruction counter value based at least in part on said modifier value; and
 - replacing the (S-N) high order bits of said permutation set of said copy with the (S-N) low order bits of said permutation set of said current instruction counter value to obtain a reference to an instruction to execute.

70. The apparatus of claim 69 wherein said modifying further comprises replacing said N low order bits of said permutation set of said copy with said modifier value.
71. The apparatus of claim 69 wherein said device is further configured to modify said current instruction counter value by adding said modifier value to said N low order bits of said permutation set of said copy.
72. The apparatus of claim 69 wherein said device is further configured to modify said current instruction counter value by subtracting said modifier value from said N low order bits of said permutation set of said copy.
73. An apparatus for application program obfuscation, the apparatus comprising an application program provider configured to:
- read a first application program comprising application program data and application program instructions;
 - determine an application program instruction location permutation that transforms said first application program into an obfuscated application program, said obfuscated application program having at least one application program instruction stored at a memory location that is based at least in part on a permutation of the memory location where the corresponding application program instruction is stored in said first application program;
 - apply said application program instruction location permutation to said first application program to create an obfuscated application program; and

send said obfuscated application program.

74. The apparatus of claim 73, said application program provider further configured to receive an application program request from a user device, said determining responsive to said receiving.

75. The apparatus of claim 73, said application program provider further configured to generate at least one instruction to fill one or more holes in the instruction stream of said application program if said instruction stream of said application program has one or more holes.

76. The apparatus of claim 73 wherein

said application program provider is configured to apply a cryptographic process to said obfuscated application program together with a cryptographic key to create an encrypted obfuscated application program after said applying said application program instruction location permutation; and
said application program provider is further configured to send said encrypted obfuscated application program.

77. A memory for storing data for access by an application program being executed on a data processing system, comprising:

a data structure stored in said memory, said data structure including information used by said application program to determine a next application program instruction to execute, said

data structure comprising one or more entries comprising a first value and a modifier value, said modifier value determining how said first value is modified, said first value matching a first one or more bits of a current application program instruction counter value.